

QuizzGame

Rogoza Calin-Andrei

15.01.2020

Introducere

Acest proiect consta in crearea unui joc de tip client/server. Jocul este reprezentat de server, iar clientii sunt utilizatorii care se conecteaza la acesta. Numarul de jucatori este nelimitat. Jucatorii conectati trebuie sa raspunda la un numar de intrebari intr-un interval de 10 secunde si raspunsurile corecte vor fi punctate. Acel utilizator care strange cel mai mare numar de puncte castiga runda. Utilizatorii care incearca sa se conecteze in timpul desfasurarii unei partide de joc vor fi repartizati intr-o alta sala de joc. Clientii se vor loga pentru a putea interactiona cu jocul.

Tehnologiile utilizate

In procesul de proiectare al aplicatiei se vor folosi:

- Limbajul C
- Server concurent TCP
- SQLite

Se va utiliza implementarea unui server TCP in limbajul C, in locul unuia de tip UDP, deoarece TCP este un protocol ce urmareste pachetele de date astfel incat nicio informatie sa nu fie pierduta sau corupta in timpul transferului, lucru ce este esential in proiectarea acestei aplicatii (raspunsurile date de jucatori trebuie sa ajunga neaparat la server, altfel, daca s-ar pierde, jocul nu s-ar mai desfasura normal).

Intrebarile si variantele de raspuns vor fi stocate intr-o baza de date SQLite. Se va utiliza acest sistem de gestiune a bazelor de date, deoarece este un sistem foarte eficient, fiabil, simplu si nu are nevoie de un server, precum alte alternative.

Arhitectura aplicatiei

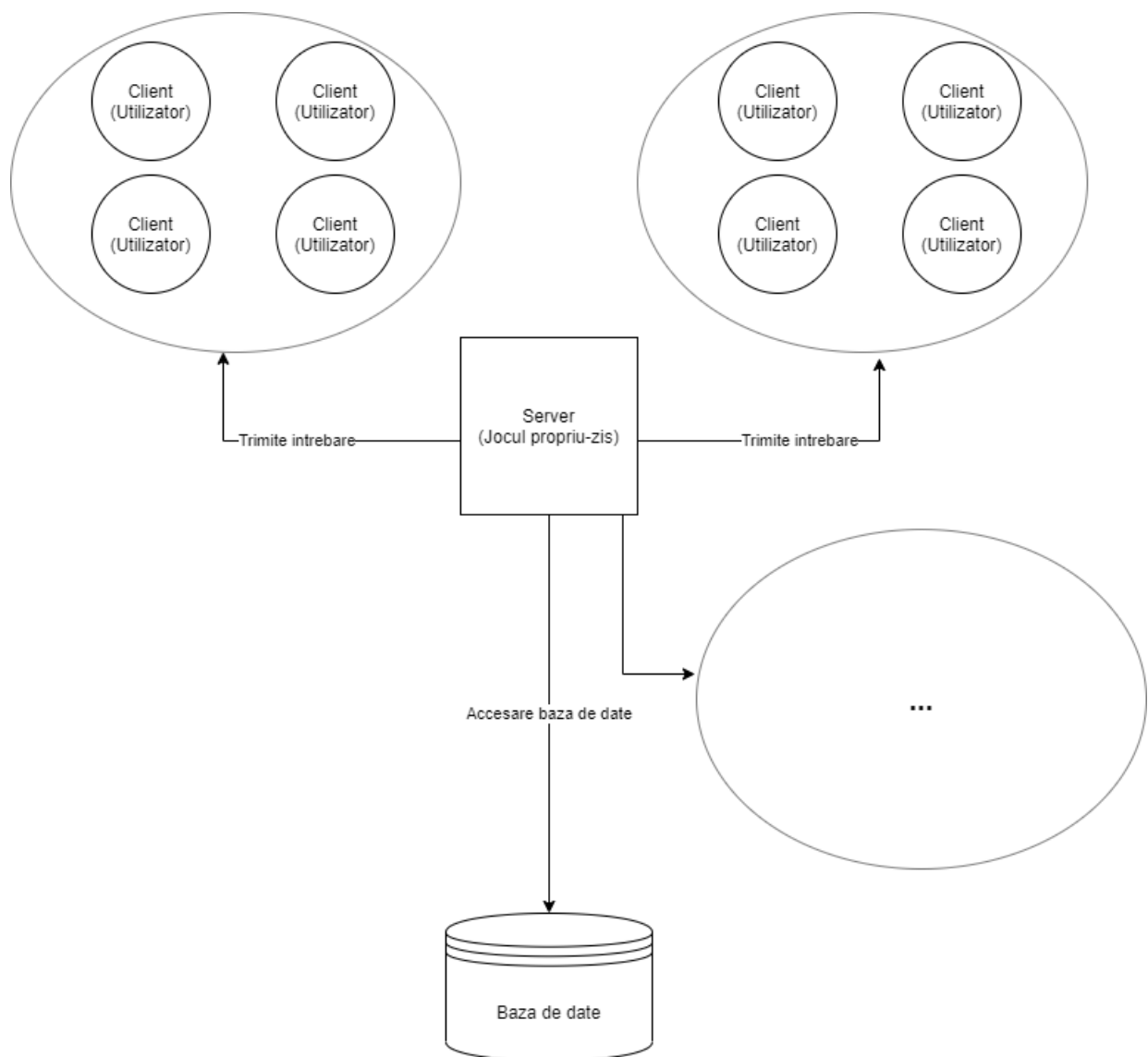


Figure 1: Diagrama proiectului (după ce clienții s-au logat și a început partida de joc)

Detalii de implementare

Aplicatia va folosi o arhitectura de tip client/server TCP concurent. Serverul va primi cereri de conectare de la clienți, numele de utilizator și parola, acestea din urmă vor fi căutate în baza de date și, dacă corespund, serverul va trimite meniul principal către client. Utilizatorul dispune în acest meniu de 3 opțiuni: căutarea unui joc, detalii cont și deconectare.

În cazul în care utilizatorul introduce tasta pentru prima opțiune, acesta va fi introdus într-o coadă de așteptare până când se vor aduna 4 jucători pentru a putea începe jocul. Zece întrebări vor fi selectate aleator din baza de date și vor fi adresate clienților, dispunând și de variante de răspuns. Utilizatorul va trebui să introducă una dintre variantele de răspuns în timp de 10 secunde și va primi ulterior confirmarea că răspunsul este corect sau greșit. În cazul în care răspunsul este corect, punctajul utilizatorului va fi incrementat cu 1. În cazul în care timpul expiră și utilizatorul nu a dat niciun răspuns, acesta va primi următoarea întrebare fără a i se

modifica punctajul. Atunci cand toti jucatorii vor termina de trimis raspunsurile la intrebari, se va afisa castigatorul pe ecran. Utilizatorii vor fi redirectionati catre meniul principal ulterior.

Altfel, daca utilizatorul alege a doua optiune, ii vor fi afisate numele de utilizator si parola pentru 5 secunde, apoi vor fi redirectionati inapoi catre meniul principal.

In ultimul caz, cand utilizatorul alege a treia optiune, acesta va fi deconectat.

Concluzii

O imbunatatire ar fi ca un jucator sa poata reveni in joc daca a pierdut dintr-un motiv oarecare conexiunea la server (revenind in joc la intrebarea curenta, si nu unde era cand a pierdut conexiunea, cummand 0 puncte pe intrebarile ce au fost puse cat timp nu era conectat). Existenta unui istoric de meciuri si a unui top pentru jucatori ar putea fi o alta optimizare bine venita.

References

- [1] Pagina cursului, <https://profs.info.uaic.ro/~computernetworks>
- [2] TCP/UDP, <https://www.geeksforgeeks.org/differences-between-tcp-and-udp/>
- [3] SQLite3 , <http://zetcode.com/db/sqlitec/>
- [4] ANSI-Colors, <https://gist.github.com/RabaDabaDoba/145049536f815903c79944599c6f952a>
- [5] Timer, <https://www.quora.com/How-do-I-put-a-time-limit-to-the-scanf-function>